0.1 7-step approach to handle modeling of unbounded domain by perfectly matched layer (Given by Mr. Debasis Mohapatra,IITR)

After implementation of PML theory the stiffness matrix, mass matrix, constitutive relation of soil , strain formulation , stress formulation got modified. To implement these modifications in finite element modeling the whole procedure has been divided in to 7-steps as given below

Step-1 Computation of system matrices mass matrix(M^e),damping matrix(C^e) and stiffness matrix(K^e)

$$\text{where } M^e_{IJ} =_{\Omega^e} \int \rho f_m N_I N_J d\Omega I \text{ , } C^e_{IJ} =_{\Omega_e} \int \rho \tfrac{c_s}{b} f_c N_I N_J d\Omega I \text{ and}$$
$$K^e_{IJ} =_{\Omega^e} \int \tfrac{\mu}{b^2} f_k N_I N_J D\Omega I$$

step-2 computation of strain at (n+1)th step

which is given by

$$\varepsilon_{n+1} = \frac{1}{\triangle t}\left[B^{\epsilon} v_{n+1} + B^{\delta} d_{n+1} + \frac{1}{\triangle t}\hat{F}^{\epsilon}\hat{\epsilon}_n - \hat{F}^{\delta}\hat{E}_n\right]$$

where v_{n+1} = velocity at (n+1) th time step , d_{n+1} = displacement at (n+1)th step

$\triangle t=$ time step in transient analysis

$\hat{\epsilon}_n$ =strain at nth time step

$E(t_{n+1}) = E_n + \epsilon_{n+1}\triangle t$ which is called accumulated strain

$$B^{\epsilon}_I = \begin{bmatrix} F^{\epsilon}_{11}N^l_{I1} & F^{\epsilon}_{21}N^l_{I1} \\ F^{\epsilon}_{12}N^l_{I2} & F^{\epsilon}_{22}N^l_{I2} \\ F^{\epsilon}_{11}N^l_{I2} + F^{\epsilon}_{12}N^l_{I1} & F^{\epsilon}_{21}N^l_{I2} + F^{\epsilon}_{22}N^l_{I1} \end{bmatrix}$$

$$N^l_{Ii} = F^l_{ij}N_{I,j},\ F^l = \left[\frac{F^e}{\triangle t} + F^p\right]^{-1},\ F^{\epsilon} = F^e F^l,\ F^{\delta} = F^p F^l$$

where $$F^e = \begin{bmatrix} 1 + f^e_1(x) & 0 \\ 0 & 1 + f^e_2(y) \end{bmatrix}, F^p = \begin{bmatrix} f^p_1(x)c_s/b & 0 \\ 0 & f^p_2(y)c_s/b \end{bmatrix}$$

By simplifying all the expressions and writing in a compacted form compatibility matrices could be written as

$$B^{\epsilon}_I = \begin{bmatrix} \dfrac{F^e_{11}N_{I,1}}{\left[\frac{F^e_{11}}{\triangle t} + F^p_{11}\right]^2} & 0 \\ 0 & \dfrac{F^e_{22}N_{I,2}}{\left[\frac{F^e_{22}}{\triangle t} + F^p_{22}\right]^2} \\ \dfrac{F^e_{11}N_{I,2}}{\left[\frac{F^e_{11}}{\triangle t} + F^p_{11}\right]\left[\frac{F^e_{22}}{\triangle t} + F^p_{22}\right]} & \dfrac{F^e_{22}N_{I,1}}{\left[\frac{F^e_{11}}{\triangle t} + F^p_{11}\right]\left[\frac{F^e_{22}}{\triangle t} + F^p_{22}\right]} \end{bmatrix}$$

B^{δ}_I is obtained by replacing $F^{\epsilon}$by F^{δ}

$$B_I^\delta = \begin{bmatrix} \frac{F_{11}^p N_{I,1}}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]^2} & 0 \\ 0 & \frac{F_{22}^p N_{I,2}}{\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]^2} \\ \frac{F_{11}^p N_{I,2}}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]} & \frac{F_{22}^p N_{I,1}}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]} \end{bmatrix}$$

$$\hat{F}^\epsilon = \begin{bmatrix} (F_{11}^\epsilon)^2 & (F_{21}^\epsilon)^2 & F_{11}^\epsilon F_{21}^\epsilon \\ (F_{12}^\epsilon)^2 & (F_{22}^\epsilon)^2 & F_{12}^\epsilon F_{22}^\epsilon \\ 2F_{11}^\epsilon F_{12}^\epsilon & 2F_{21}^\epsilon F_{22}^\epsilon & F_{11}^\epsilon F_{22}^\epsilon + F_{12}^\epsilon F_{21}^\epsilon \end{bmatrix}$$

by simplifying all the expressions

$$\hat{F}^\epsilon = \begin{bmatrix} \left(\frac{F_{11}^e}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]}\right)^2 & 0 & 0 \\ 0 & \left(\frac{F_{22}^e}{\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]}\right)^2 & 0 \\ 0 & 0 & \left(\frac{F_{11}^e}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]}\right)\left(\frac{F_{22}^e}{\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]}\right) \end{bmatrix}$$

similarly

$$\hat{F}^\delta = \begin{bmatrix} (F_{11}^\delta)^2 & (F_{21}^\delta)^2 & F_{11}^\delta F_{21}^\delta \\ (F_{12}^\delta)^2 & (F_{22}^\delta)^2 & F_{12}^\delta F_{22}^\delta \\ 2F_{11}^\delta F_{12}^\delta & 2F_{21}^\delta F_{22}^\delta & F_{11}^\delta F_{22}^\delta + F_{12}^\delta F_{21}^\delta \end{bmatrix}$$

$$\hat{F}^\delta = \begin{bmatrix} \left(\frac{F_{11}^p}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]}\right)^2 & 0 & 0 \\ 0 & \left(\frac{F_{22}^p}{\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]}\right)^2 & 0 \\ 0 & 0 & \left(\frac{F_{11}^p}{\left[\frac{F_{11}^e}{\triangle t}+F_{11}^p\right]}\right)\left(\frac{F_{22}^p}{\left[\frac{F_{22}^e}{\triangle t}+F_{22}^p\right]}\right) \end{bmatrix}$$

step-3 computation of internal stress vector at $(n+1)th$ time step

which is given by expression

$$\hat{\sigma}_{n+1} = \left(1 + \frac{2\zeta b}{c_s \triangle t}\right) D\hat{\varepsilon}_{n+1} - \frac{2\zeta b}{c_s \triangle t} D\hat{\varepsilon}_n$$

where $\hat{\varepsilon}_{n+1}$ and $\hat{\varepsilon}_n$could be calculated from step-2.

$$D = \textbf{constitutive matrix} = \begin{bmatrix} K+4G/3 & K-2G/3 & 0 \\ K-2G/3 & K+4G/3 & 0 \\ 0 & 0 & G \end{bmatrix}$$

step-4 computation of internal force P_{n+1}^ewhich is given by the expression

$$P_{n+1}^e =_{\Omega_e} \int \bar{B}^T \sigma_{n+1} d\Omega \, + \, _{\Omega^e} \int \bar{B}^{PT} \sum\nolimits_n$$

σ_{n+1} is calculated from step-3.

$\sum(t_{n+1})$ = accumulated stress $= \sum_n + \sigma_{n+1} \triangle t$

where $\bar{B} = \bar{B}^e + \triangle t \bar{B}^p$

$$\bar{B_I}^e = \begin{bmatrix} N^e_{I1} & 0 \\ 0 & \bar{N_{I2}}^e \\ \bar{N_{I2}}^e & \bar{N_{I1}}^e \end{bmatrix}, \bar{B_I}^p = \begin{bmatrix} \bar{N_{I1}}^p & 0 \\ 0 & \bar{N_{I2}}^p \\ \bar{N_{I2}}^p & \bar{N_{I1}}^p \end{bmatrix}$$

with $\bar{N}^e = \bar{F_{ij}}^e N_{I,j}$and $\bar{N_{Ii}}^p = \bar{F_{ij}}^p N_{I,j}$

$$\bar{F}^e = \begin{bmatrix} 1 + f^e_2(y) & 0 \\ 0 & 1 + f^e_1(x) \end{bmatrix} \text{ and } \bar{F}^p = \begin{bmatrix} f^p_2(y)c_s/b & 0 \\ 0 & f^p_1(x)c_s/b \end{bmatrix}$$

step-5 computation of tangent matrices c , k

$c =$ **damping tangent matix**$=_{\Omega^e} \int \bar{B}^T \bar{D} B^\epsilon d\Omega$

k =stiffness tangent matrix $=_{\Omega^e} \int \bar{B}^T \bar{D} B^\epsilon d\Omega$

$\bar{D} = \frac{1}{\triangle t} \left(1 + \frac{2\zeta b}{c_s \triangle t}\right) D$

step-6 computation of effective internal force $\bar{P^e}_{n+1}$

$\bar{P^e}_{n+1} = M^e a_{n+1} + C^e v_{n+1} + K^e d_{n+1} + P^e_{n+1}$

where M^e, C^e and K^e are calculated from Step-1

$P^e_{n+1}$is calculated from step-4

a_{n+1} =acceleration at $(n+1)th$ step.

step-7 computation of tangent stiffness matrix

$\bar{K}^e = \alpha_k(K^e + k) + \alpha_c(C^e + c) + \alpha_m M^e$

where $\alpha_k = 1$, $\alpha_c = \frac{\gamma}{\beta \triangle t}$, $\alpha_m = \frac{1}{\beta \triangle t^2}$ for Newmark's time integration method.

for average acceleration method (γ=0.5, β=0.25)

for linear acceleration method ($\gamma = 0.5$, $\beta = 0.167$)

$\bar{K}^e$**is** independent of solution thus it has to be computed only once. However the internal force $P^e_{n+1}$is dependent on previous time steps, so it is needed to be computed for each time step.

0.2 Introduction of new material to opensees

0.2.1 Soil

For this research work a new material named as soil has been added to open sees frame work. Soil is a material which can only be used for quad element of plainstrain type. This new material soil considers kelvin-voigt material damping in time domain. Whose constitutive relation is given as
which is given by expression

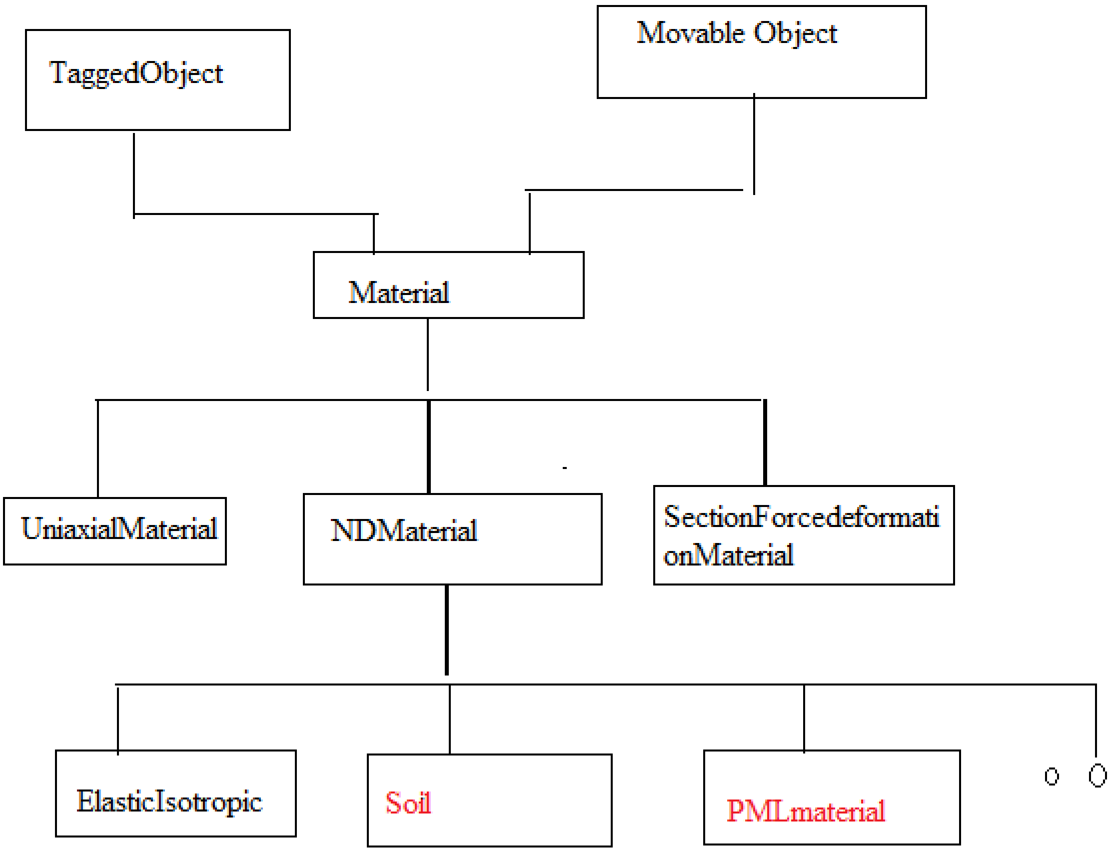

Figure 0.2.1: Material class hierarchy in Opensees Framework

$\hat{\sigma}_{n+1} = \left(1 + \frac{2\zeta b}{c_s \triangle t}\right) D\hat{\varepsilon}_{n+1} - \frac{2\zeta b}{c_s \triangle t} D\hat{\varepsilon}_n$

$$D = \textbf{constitutive matrix} = \begin{bmatrix} K+4G/3 & K-2G/3 & 0 \\ K-2G/3 & K+4G/3 & 0 \\ 0 & 0 & G \end{bmatrix}$$

Diagram given below shows class hierarchy of opensees material model

opensees command used to access this material is given by

nDMaterial soil $tag $K $G $rho $b $jita $T

tag unique material tag used for modeling

K bulk modulus of soil

G shear modulus of soil

rho Mass density of soil

b characteristic length of footing

jita kelvin voigt damping ratio of soil

T time step of analysis

0.2.2 PMLmaterial

Similarly PMLmaterial is a NDmaterial which can only be accessed by PMLelement. It has one more functions than the general nDmaterial abstract class which is given by *getStrainsum(). It* has same constitutive relation as soil material

nDMaterial PMLmaterial $tag $K $G $rho $b $jita $T

tag unique material tag used for modeling

K bulk modulus of soil

G shear modulus of soil

rho Mass density of soil

b characteristic length of footing

jita kelvin voigt damping ratio of soil

T time step of analysis

0.3 Introduction of New element in opensees

Element class in opensees is a abstract class which has many derived classes. For 2-D four noded finite element there is a class quad which could be used for both plane strain and plane stress elements.The newly added element class PMLelement is a similar four noded element class whose some functions are modified to meet theoretical expressions of PML.Here are some important functions that are modified.

1. **double PMLelement::shapeFunction(double xi, double eta)**
2. **int PMLelement::update()**
3. **const Matrix& PMLelement::getTangentStiff()**
4. **const Matrix& PMLelement::getInitialStiff()**
5. **const Matrix& PMLelement::getMass()**
6. **const Matrix& PMLelement::getDamp()**
7. **int PMLelement::addInertiaLoadToUnbalance(const Vector &accel)**
8. **const Vector& PMLelement::getResistingForce()**

An important modification has been made to impliment NDmaterial PMLmaterial specifically for PMLelement. that is done by modifying the general coding as given below

```
theMaterial = new PMLmaterial *[4];
if (theMaterial == 0) { opserr << "PMLelement::PMLelement - failed allocate material model pointer\n";
exit(-1); }
NDMaterial **theMatcopy;
theMatcopy=new NDMaterial *[4];
for (int i = 0; i < 4; i++) {
theMatcopy[i]=m.getCopy("PMLmaterial");
if (theMatcopy[i] !=0)
{ theMaterial[i] = (PMLmaterial *)theMatcopy[i]; }
else { opserr << "PMLmaterial::PMLmaterial - failed allocate material model pointer\n"; } }
if (theMaterial == 0)
{ opserr << "PMLelement::PMLelement - failed allocate material model pointer\n";
exit(-1); }
```

Tcl command used to access PMLelement in opensees is given as

element PMLelement $elementtag $node1 $node2 $node3 $node4 $thickness "PMLmaterial" $mattag $b $G $rho $L $H $jita L_{PML}$ $T $C $n

elementtag unit element tag to define PMLelement in modelling

node1,node2,node3,node4 node numbers which fournoded PMLelement connects. Node numbers should be introduced in anticlockwise direction.

thickness thickness of planestrain element. Generally it is considered as 1.0.

mattag Material tag of PMLmaterial

b characteristic length of footing

G shear modulus of soil

rho mass density of soil

L length of linear part of soil considered in X-direction which is measured from center of footing

H length of linear part of soil considered in Y-direction which is also needed to be measured from center of footing

jita Kelvin Voigt damping ratio of soil

L$_{PML}$ Thickness of PML in both X and Y direction. However coding is developed in such a way that the thickness of PML in both X and Y direction are taken to be same.

T time step used for analysis

C attenuation strength

n exponent of attenuation function

attenuation functions used for this coding in X-direction is given as $f^p(x) = f^e(x) = C\left(\frac{x-L}{L_{PML}}\right)^n$
attenuation functions used for this coding in Y-direction is given as $f^p(y) = f^e(y) = C\left(\frac{y-H}{L_{PML}}\right)^n$